**Listing of Claims:**

1. (Currently Amended) A method for adapting the format of a message, comprising:

mapping a plurality of fields in a first message having a first native message format with corresponding fields in a first structured event message in a structured event ~~message~~ format;

mapping a plurality of fields in a second message having a second native message format with corresponding fields in a second structured event message in the structured event format.

2. (Original) The method of claim 1 wherein the first native message format is a JMS message format.

3. (Original) The method of claim 2 wherein the native JMS format is selected from the group consisting of TextMessage, BytesMessage; MapMessage; ObjectMessage; and StreamMessage.

4. (Original) The method of claim 3 wherein the fields in a message in a JMS MapMessage format are mapped onto the fields in a structured event format.

5. (Original) The method claim 4 wherein the domain name and type name properties of a JMS-based message are concatenated and placed in the JMSType field of a JMS MapMessage.

6. (Original) The method of claim 5 wherein the domain name portion of the concatenated JMSType field of a JMS MapMessage is mapped onto the domain_name field of a structured event.

7. (Original) The method of claim 6 further comprising the mapping of the type name portion of said concatenated JMSType field onto the type_name field of said structured event.

8. (Cancelled) The method of claim 7 further comprising the mapping of the JMSMessageID field in said JMS MapMessage onto a corresponding field in the variable header section of said structured event.

9. (Cancelled) The method of claim 8 further comprising the mapping of the JMSTimestamp field in said JMS MapMessage onto a corresponding field in the variable header section of said structured event.

10. (Cancelled) The method of claim 9 further comprising the mapping of the JMSDeliveryMode field in said JMS MapMessage onto a corresponding field in the variable header section of said structured event.

11. (Cancelled) The method of claim 10 further comprising the mapping of the properties section in said JMS MapMessage onto the filterable body section of said structured event.

12. (Currently Amended) The method of claim 7 ~~11~~ further comprising the mapping of the body section in said JMS MapMessage onto the remaining body section of said structured event.

13. (Original) The method of claim 3 wherein the fields in a message in a structured event format are mapped onto the fields in a JMS MapMessage format.

14. (Original) The method claim 13 wherein the domain_name and type_name fields of a structured event are concatenated and the concatenated field is mapped onto the JMSType field of a JMS MapMessage.

15. (Original) The method of claim 14 further comprising the mapping of the event_type field in said structured event onto the JMSCorrelationID field in said JMS MapMessage.

16. (Original) The method of claim 15 further comprising the mapping of a timestamp field in the variable header section of said structured event onto a corresponding field in the properties section of said JMS MapMessage.

17. (Original) The method of claim 16 further comprising the mapping of the filterable body section of said structured event onto the properties section in said JMS MapMessage.

18. (Original) The method of claim 17 further comprising the mapping of the remaining body section of said structured event onto the body section in said JMS MapMessage.

19. (Currently Amended) The method of claim 1 wherein the second native message format is a mainframe message format.

20. (Original) The method of claim 19 wherein the native mainframe format is a Cobol copybook.

21. (Original) The method of claim 20 wherein the fields in a message in a copybook format are mapped onto the fields in a structured event format.

22. (Original) The method of claim 21 wherein a field of said copybook containing the domain name is mapped onto the domain_name field of a structured event.

23. (Original) The method of claim 22 further comprising the mapping of a field of said copybook containing the type name onto the type_name field of said structured event.

24. (Original) The method of claim 23 further comprising the mapping of a field of said copybook containing the event name onto the event_name field of said structured event.

25. (Original) The method of claim 24 further comprising the mapping of a freeform data field in said copybook onto the remaining body section of said structured event.

26. (Original) The method of claim 20 wherein the fields in a message in a structured event format are mapped onto the fields in a copybook format.

27. (Original) The method of claim 26 wherein the domain_name field of said structured event is mapped onto a corresponding field of said copybook.

28. (Original) The method of claim 27 further comprising the mapping of the type_name field of said structured event onto a corresponding field of said copybook.

29. (Original) The method of claim 28 further comprising the mapping of the event_name field of said structured event onto a corresponding field of said copybook.

30. (Original) The method of claim 29 further comprising the mapping of the data in the filterable body section of said structured event field onto a freeform data field in said copybook.

31. (New) The method of claim 2 wherein the second native message format is a mainframe message format.

32. (New) A method for brokering messages between middleware systems comprising:

communicating a message from a mainframe system in a Cobol copybook format;

mapping the message in Cobol copybook format onto the fields in a structured event format;

communicating the message converted from Cobol copybook format to structured event format to a middleware brokering system;

communicating a message from a JMS system in a JMS message format;

mapping the message in the JMS format onto the fields in a structured event format;

communicating the message converted from JMS format to the structured event format to the middleware brokering system;

communicating a message from a CORBA system in the structured event format to the middleware brokering system;

using the middleware broker to determine the destination for each of the messages from the JMS, CORBA, and mainframe systems; and

directing each of the messages to the appropriate one of the JMS, CORBA, and mainframe systems.

33. (New) The method of claim 32 further comprising:

converting the messages destined for the mainframe system from the structured event format to Cobol copybook format; and

converting the messages destined for the JMS system from structured event format to the JMS system.

34. (New) The method of claim 32 further comprising:

registering each of the messages with a publish/subscribe engine; and

brokering the messages between the JMS, CORBA, and mainframe systems based on the messages that the JMS, CORBA, and mainframe systems have registered to receive.